



智慧型手機程式設計

A6409001_資工 3B_邱郁涵

目錄

內容

一、 題目	3
題目：生活小幫手 APP 設計-使用警示對話方塊	3
二、 實作結果	4
手機展示畫面	4
三、 程式碼	5
XML 檔	5
JAVA 檔	12
三、 學習心得	22
心得感想	22

圖目錄

圖 1	4
圖 2	7
圖 3	9
圖 4	11
圖 5	12
圖 6	15
圖 7	18
圖 8	21

一、題目

題目：生活小幫手 App 設計-使用警示對話方塊

題目說明：利用課堂所學，並參考老師提供的基於警示對話方塊之簡易記帳簿 App，設計一支生活小幫手 App。App 的功能用途由你自己訂定，參考題目如下：(1)飲食記錄 App-可紀錄與查詢日常飲食項目與卡洛里量、(2)體重記錄 App-可紀錄與查詢日常的體重與 BMI、(3)運動記錄 App-可記錄與查詢日常的運動項目、運動時間與所消耗的卡洛里量、...等。

二、實作結果

手機展示畫面



圖 1

實作影片___:

<https://drive.google.com/open?id=1hnCdnC3lYv13NrKfIXa4TmTOA958GSkA>

三、程式碼

XML 檔

Search or jump to... / Pull requests Issues Marketplace Explore

QINGSHURUNINDEXINGMING / PccuHW

Watch 0 Star 0 Fork 0

Code Issues Pull requests Projects Wiki Insights Settings

Branch: master PccuHW / 三下 / 智慧型手機程式設計 / Tallybook1_AlertDialogs / app / src / main / res / layout / activity_main.xml Find file Copy path

QINGSHURUNINDEXINGMING 2019/05/05 17:24 a56faab · an hour ago 1 contributor

182 lines (150 sloc) · 5.69 KB Raw Blame History

```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
4     android:id="@+id/outerScroller"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7
8     <LinearLayout
9         android:layout_width="match_parent"
10        android:layout_height="match_parent"
11        android:orientation="vertical">
12
13        <LinearLayout
14            android:layout_width="match_parent"
15            android:layout_height="wrap_content"
16            android:orientation="horizontal">
17
18            <TextView
19                android:id="@+id/Thumbnail"
20                android:layout_width="wrap_content"
21                android:layout_height="wrap_content"
22                android:text="今日步數:"
23                android:textSize="16sp"
24                android:textColor="#000000" //
25
26
27            <EditText
28                android:id="@+id/ETNumber"
29                android:layout_width="wrap_content"
30                android:layout_height="wrap_content"
31                android:layout_weight="0.8"
32                android:hint="請輸入步數"
33                android:inputType="number"
34                android:textSize="16sp" //
35
36        </LinearLayout>
37
38        <LinearLayout
39            android:layout_width="match_parent"
40            android:layout_height="wrap_content">
41
42            <TextView
43                android:id="@+id/txtType"
44                android:layout_width="wrap_content"
45                android:layout_height="wrap_content"
46                android:text="運動型態:"
47                android:textSize="16sp"/>
48
49            <Button
50                android:id="@+id/btnChoseType"
51                android:layout_width="wrap_content"
52                android:layout_height="wrap_content"
53                android:text="選擇運動型態"
54                android:textSize="16sp"
55                android:layout_weight="0.4"
56                android:textColor="#f000ff"
57                android:onClick="btnChoseType_Click"/>
58
59            <Button
60                android:id="@+id/btnManageExpenseType"
61                android:layout_width="wrap_content"
62                android:layout_height="wrap_content"
63                android:text="管理運動型態"
64                android:textSize="16sp"
65                android:layout_weight="0.3"
66                android:textColor="#0000ff"
67                android:onClick="btnManageExpenseType_Click"/>
68
69        </LinearLayout>
70
```

```

40
41 <LinearLayout
42     android:layout_width="match_parent"
43     android:layout_height="wrap_content" >
44
45     <TextView
46         android:id="@+id/txtTitle"
47         android:layout_width="wrap_content"
48         android:layout_height="wrap_content"
49         android:text="日期"
50         android:textSize="16sp"/>
51
52     <EditText
53         android:id="@+id/edtComment"
54         android:layout_width="match_parent"
55         android:layout_height="wrap_content"
56         android:hint="添加评论(可不填)"
57         android:textSize="16sp"
58         android:layout_weight="0.8"/>
59 </LinearLayout>
60
61 <LinearLayout
62     android:layout_width="match_parent"
63     android:layout_height="wrap_content" >
64
65     <Button
66         android:id="@+id/btnDate"
67         android:layout_width="wrap_content"
68         android:layout_height="wrap_content"
69         android:text="日期"
70         android:textSize="16sp"
71         android:onClick="btnDate_Click"
72         android:textColor="#000000"/>
73
74     <TextView
75         android:id="@+id/txtDate"
76         android:layout_width="wrap_content"
77         android:layout_height="wrap_content"
78         android:hint="日期"
79         android:layout_weight="0.8"
80         android:textSize="16sp"/>
81 </LinearLayout>
82
83 <LinearLayout
84     android:layout_width="match_parent"
85     android:layout_height="wrap_content" >
86
87     <Button
88         android:id="@+id/btnSave"
89         android:layout_width="match_parent"
90         android:layout_height="wrap_content"
91         android:text="保存"
92         android:textSize="16sp"
93         android:onClick="btnSave_Click"/>
94 </LinearLayout>
95
96 <LinearLayout
97     android:layout_width="match_parent"
98     android:layout_height="wrap_content" >
99
100     <Button
101         android:id="@+id/btnQuery"
102         android:layout_width="match_parent"
103         android:layout_height="wrap_content"
104         android:text="查询"
105         android:textSize="16sp"
106         android:onClick="btnQuery_Click"/>
107 </LinearLayout>
108
109 <LinearLayout
110     android:layout_width="match_parent"
111     android:layout_height="wrap_content" >
112
113     <ScrollView
114         android:id="@+id/scrollView1"
115         android:layout_width="match_parent"
116         android:layout_height="220px">
117
118         <TextView
119             android:id="@+id/txtMainOutput"
120             android:layout_width="match_parent"
121             android:layout_height="match_parent"/>
122     </ScrollView>
123 </LinearLayout>
124
125 <LinearLayout
126     android:layout_width="match_parent"
127     android:layout_height="wrap_content" >
128
129     <Button
130         android:id="@+id/btnClearMainOutput"
131         android:layout_width="wrap_content"
132         android:layout_height="wrap_content"
133         android:layout_weight="0.5"
134         android:onClick="btnClearMainOutput_Click"
135         android:text="清除主输出"
136         android:textColor="#000000"/>
137
138     <Button
139         android:id="@+id/btnCloseApp"
140         android:layout_width="wrap_content"
141         android:layout_height="wrap_content"
142         android:layout_weight="0.5"
143         android:onClick="btnCloseApp_Click"
144         android:text="关闭应用"
145         android:textColor="#FF0000"/>
146 </LinearLayout>
147 </ScrollView>

```




Search or jump to...

Pull requests Issues Marketplace Explore



QINGSHURUNINDEXINGMING / PccuHW

Watch 0 Star 0 Fork 0

Code Issues Pull requests Projects Wiki Insights Settings

Branch: master PccuHW / 三下 / 智慧型手機程式設計 / Tallybook1_AlertDialogs / app / src / main / res / layout / manageexpensetypedialog.xml Find file Copy path

QINGSHURUNINDEXINGMING 2019/05/05 17:24 a56faab · an hour ago 1 contributor

99 lines (82 sloc) 3.21 KB Raw Blame History

```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
4     android:id="@+id/outerScroller"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7
8     <LinearLayout
9         android:layout_width="match_parent"
10        android:layout_height="match_parent"
11        android:orientation="vertical">
12
13        <ScrollView
14            android:id="@+id/innerScroller"
15            android:layout_width="fill_parent"
16            android:layout_height="350px">
17
18            <TextView
19                android:id="@+id/txtShowExpenseTypes"
20                android:layout_width="match_parent"
21                android:layout_height="match_parent"/>
22
23        </ScrollView>
24
25        <LinearLayout
26            android:layout_width="match_parent"
27            android:layout_height="wrap_content">
28
29            <EditText
30                android:id="@+id/etNewExpenseType"
31                android:layout_width="wrap_content"
32                android:layout_height="wrap_content"
33                android:ems="11"
34                android:hint="請輸入新增動支型態"
35                android:textSize="12sp"
36                android:inputType="text"
37                android:layout_weight="0.7"/>
38
39            <Button
40                android:id="@+id/btnCreateNewExpenseType"
41                android:layout_width="wrap_content"
42                android:layout_height="wrap_content"
43                android:layout_weight="0.3"
44                android:text="新增"
45                android:textColor="#E60000"
46                android:onClick="btnCreateNewExpenseType_Click"/>
47
48        </LinearLayout>
49
50        <LinearLayout
51            android:layout_width="match_parent"
52            android:layout_height="wrap_content">
53
54            <EditText
55                android:id="@+id/etOldExpenseType"
56                android:layout_width="wrap_content"
57                android:layout_height="wrap_content"
58                android:layout_weight="0.7"
59                android:ems="11"
60                android:hint="請輸入要刪除的動支型態編號"
61                android:textSize="12sp"
62                android:inputType="numberDecimal"/>
63
```

```

64         <Button
65             android:id="@+id/btnDeleteExpenseType"
66             android:layout_width="wrap_content"
67             android:layout_height="wrap_content"
68             android:layout_height="0.3"
69             android:text="删除"
70             android:textColor="#ff0000"
71             android:onClick="btnDeleteExpenseType_Click"/>
72     </LinearLayout>
73
74     <LinearLayout
75         android:layout_width="match_parent"
76         android:layout_height="wrap_content" >
77
78         <Button
79             android:id="@+id/btnDisplayExpenseTypes"
80             android:layout_width="wrap_content"
81             android:layout_height="wrap_content"
82             android:layout_height="0.45"
83             android:text="显示显示全部类型"
84             android:textColor="#0000ff"
85             android:onClick="btnDisplayExpenseTypes_Click"/>
86
87         <Button
88             android:id="@+id/btnCloseDialog"
89             android:layout_width="wrap_content"
90             android:layout_height="wrap_content"
91             android:layout_height="0.35"
92             android:text="隐藏对话框"
93             android:textColor="#ff00ff"
94             android:onClick="btnCloseDialog_Click"/>
95     </LinearLayout>
96
97 </LinearLayout>
98 </ScrollView>

```

圖 3

Search or jump to...

Pull requestsIssuesMarketplaceExplore

QINGSHURUNINDEXINGMING / PccuHW

Watch0Star0Fork0

CodeIssuesPull requestsProjectsWikiInsightsSettings

Branch: masterPccuHW / 三下 / 智慧型手機程式設計 / Tallybook1_AlertDialogs / app / src / main / res / layout / queryhistoricalexpensdialog.xmlFind fileCopy path

QINGSHURUNINDEXINGMING 2019/05/05, 17:24

as6faab · an hour ago

1 contributor

156 lines (127 sloc)4.93 KB

RawBlameHistory

```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
4     android:id="@+id/outterScroller"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7
8     <LinearLayout
9         android:layout_width="match_parent"
10        android:layout_height="match_parent"
11        android:orientation="vertical">
12
13        <LinearLayout
14            android:layout_width="match_parent"
15            android:layout_height="wrap_content">
16
17            <Button
18                android:id="@+id/btnChoseStartDate"
19                android:layout_width="wrap_content"
20                android:layout_height="wrap_content"
21                android:onClick="btnChoseStartDate_Click"
22                android:text="起始日期"
23                android:textColor="#FF0000"
24                android:textSize="14sp" />
25
26
27        <TextView
28            android:id="@+id/txtStartDate"
29            android:layout_width="wrap_content"
30            android:layout_height="wrap_content"
31            android:layout_weight="0.4"
32            android:gravity="center"
33            android:hint="點擊左邊取得起始日期"
34            android:textSize="12sp"/>
35
36        <Button
37            android:id="@+id/btnChoseEndDate"
38            android:layout_width="wrap_content"
39            android:layout_height="wrap_content"
40            android:onClick="btnChoseEndDate_Click"
41            android:text="結束日期"
42            android:textSize="14sp"
43            android:textColor="#FF0000"/>
44
45        <TextView
46            android:id="@+id/txtEndDate"
47            android:layout_width="wrap_content"
48            android:layout_height="wrap_content"
49            android:layout_weight="0.4"
50            android:gravity="center"
51            android:hint="點擊左邊取得結束日期"
52            android:textSize="12sp"/>
53    </LinearLayout>
54
55    <LinearLayout
56        android:layout_width="match_parent"
57        android:layout_height="wrap_content">
58
59        <RadioGroup
60            android:id="@+id/radioGroup1"
61            android:layout_width="match_parent"
62            android:layout_height="wrap_content"
63            android:orientation="horizontal">
64
65            <RadioButton
66                android:id="@+id/radioButtonByDate"
67                android:layout_width="wrap_content"
68                android:layout_height="wrap_content"
69                android:checked="true"
70                android:text="按日期查詢顯示"
71                android:textSize="12sp"
72                android:layout_height="0.5"/>
73
74            <RadioButton
75                android:id="@+id/radioButtonByType"
76                android:layout_width="wrap_content"
77                android:layout_height="wrap_content"
78                android:text="按運動型態顯示"
79                android:textSize="12sp"
80                android:layout_height="0.5"/>
81        </RadioGroup>
82    </LinearLayout>
83
84    <LinearLayout
85        android:layout_width="match_parent"
86        android:layout_height="wrap_content">
87
88
```

10

```

93         <Button
94             android:id="@+id/btnQuery"
95             style="@android:attr/buttonStyleSmall"
96             android:layout_width="wrap_content"
97             android:layout_height="wrap_content"
98             android:layout_weight="0.5"
99             android:onClick="btnQuery_Click"
100             android:text="資料查詢功能"
101             android:textSize="16sp" />
102
103         <Button
104             android:id="@+id/btnDelete"
105             style="@android:attr/buttonStyleSmall"
106             android:layout_width="wrap_content"
107             android:layout_height="wrap_content"
108             android:layout_weight="0.5"
109             android:onClick="btnDelete_Click"
110             android:text="刪除運動狀態"
111             android:textSize="16sp" />
112     </LinearLayout>
113
114     <LinearLayout
115         android:layout_width="match_parent"
116         android:layout_height="wrap_content">
117
118         <ScrollView
119             android:id="@+id/innerScroller"
120             android:layout_width="match_parent"
121             android:layout_height="350px">
122
123             <TextView
124                 android:id="@+id/txtMainOutput"
125                 android:layout_width="match_parent"
126                 android:layout_height="match_parent"/>
127
128         </ScrollView>
129     </LinearLayout>
130
131
132     <LinearLayout
133         android:layout_width="match_parent"
134         android:layout_height="wrap_content">
135
136         <Button
137             android:id="@+id/btnClearMainOutput"
138             android:layout_width="wrap_content"
139             android:layout_height="wrap_content"
140             android:layout_weight="0.5"
141             android:onClick="btnClearMainOutput_Click"
142             android:text="清除顯示區"
143             android:textColor="#000000"/>
144
145         <Button
146             android:id="@+id/btnCloseDialog"
147             android:layout_width="wrap_content"
148             android:layout_height="wrap_content"
149             android:layout_weight="0.5"
150             android:onClick="btnCloseDialog_Click"
151             android:text="關閉對話方塊"
152             android:textColor="#FF00FF"/>
153     </LinearLayout>
154
155 </LinearLayout>
156 </ScrollView>

```

圖 4

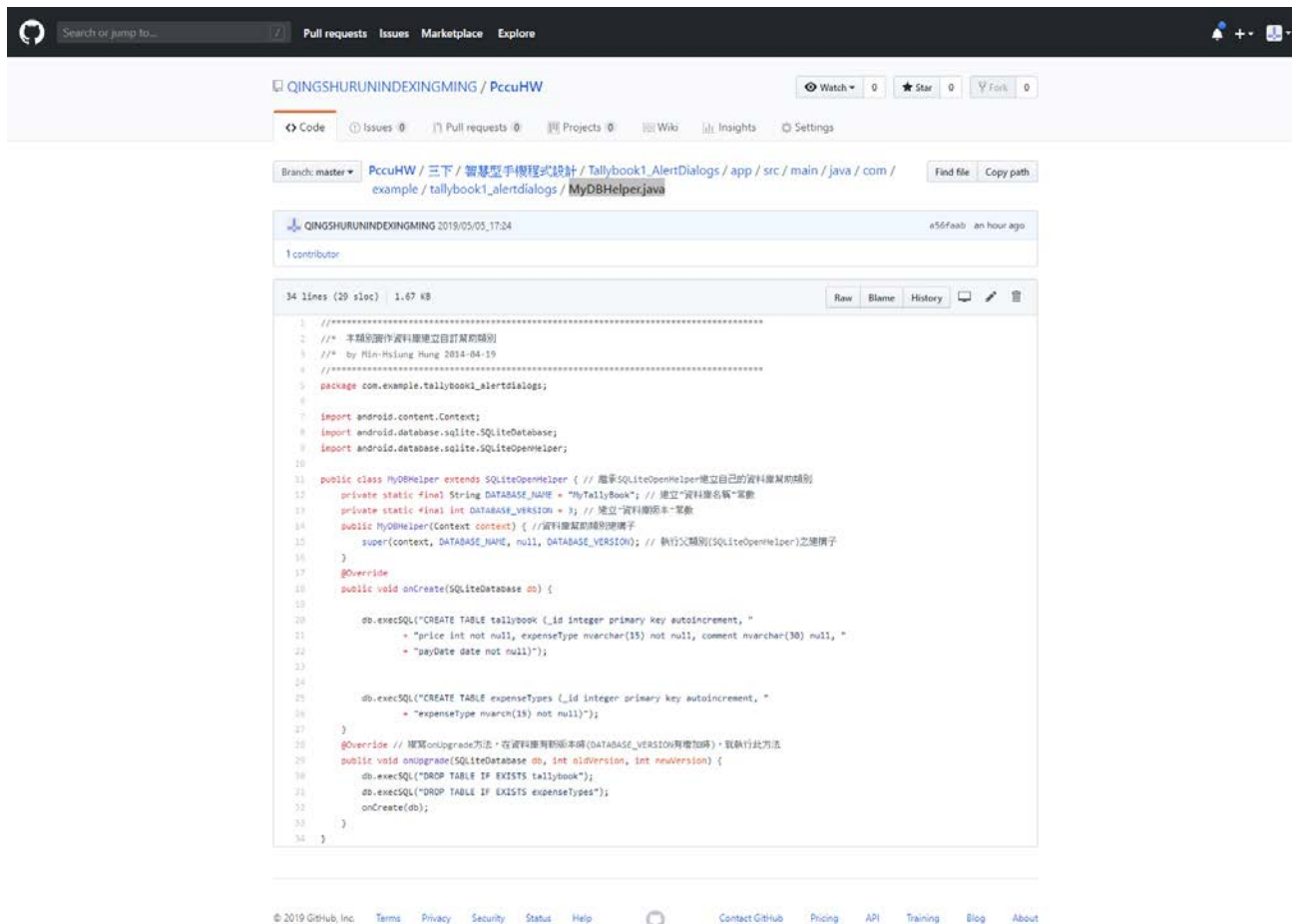


圖 5



Search or jump to...

Pull requests Issues Marketplace Explore



QINGSHURUNINDEXINGMING / PccuHW

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Branch: master PccuHW / 三下 / 智慧型手機程式設計 / Tallybook1_AlertDialogs / app / src / main / java / com / example / tallybook1_alertdialogs / MainActivity.java

Find file Copy path

QINGSHURUNINDEXINGMING 2019/05/05, 17:24

a56fab8 · an hour ago

1 contributor

261 lines (230 sloc) | 9.65 KB

Raw Blame History

```
1  //*****
2  /** 本類別實作簡易記錄簿主活動內之各個方法
3  /**  by Min-Hsiung Hung 2014-04-19
4  //*****
5  package com.example.tallybook1_alertdialogs;
6
7  import android.app.Activity;
8  import android.app.AlertDialog;
9  import android.app.DatePickerDialog;
10 import android.app.TimePickerDialog;
11 import android.content.Context;
12 import android.content.DialogInterface;
13 import android.content.Intent;
14 import android.database.Cursor;
15 import android.database.sqlite.SQLiteDatabase;
16 import android.os.Bundle;
17 import android.view.View;
18 import android.widget.AdapterView;
19 import android.widget.AdapterView.OnItemClickListener;
20 import android.widget.ArrayAdapter;
21 import android.widget.Button;
22 import android.widget.DatePicker;
23 import android.widget.EditText;
24 import android.widget.ImageButton;
25 import android.widget.RadioButton;
26 import android.widget.ScrollView;
27 import android.widget.TextView;
28 import android.widget.TimePicker;
29 import android.widget.Toast;
30 import android.app.DatePickerDialog;
31 import java.util.Calendar;
32 import java.util.Date;
33 import java.sql.*;
34 import java.text.DateFormat;
35 import java.text.ParseException;
36 import java.text.SimpleDateFormat;
37
38 import com.example.tallybook1_alertdialogs.ManageExpenseTypeDialog;
39 import com.example.tallybook1_alertdialogs.QueryHistoricalExpenseDialog;
40
41 public class MainActivity extends Activity
42 {
43     // 與建立資料庫有關之變數宣告
44     private static String DATABASE_NAME = "tallybook"; //宣告"儲款資料庫"名稱
45     private static String DATABASE_TABLE1 = "expenseTypes"; //宣告"儲款型態資料庫"名稱
46     private SQLiteDatabase db; //宣告資料庫變數
47     private DBHelper dbHelper; //宣告資料庫幫助變數
48
49
50
51     // 與顯示介面元件有關之變數宣告
52     private EditText etxtPrice, etxtComment;
53     private Button btnManageExpenseType, btnChoseType;
54     private Button btnDate, btnSave, btnQuery, btnDelete;
55     private Button btnChoseStartDate, btnChoseEndDate;
56     private Button btnCloseApp;
57     private Button btnClearMainOutput;
58     private TextView txtMainOutput, txtDate;
59     private TextView txtStartDate, txtEndDate;
60
61     // 其他變數宣告
62     String [] initialExpenseTypes = {"慢走", "慢跑", "快走", "快跑", "其他"};
63     String [] expenseTypes;
64
65
66     int Price;
67     String ExpenseType="";
68     String Comment="";
69     String PayDateString="";
70
71     int Year, Month, Day;
72
73
74     @Override
75     public void onCreate(Bundle savedInstanceState)
76     {
77         super.onCreate(savedInstanceState);
78         setContentView(R.layout.activity_main); //顯示操作界面
79         // 取得應用程式介面之上所有元件
80         etxtPrice=(EditText) findViewById(R.id.etxtPrice);
81         etxtComment=(EditText) findViewById(R.id.etxtComment);
82         btnManageExpenseType = (Button) findViewById(R.id.btnManageExpenseType);
83         btnChoseType = (Button) findViewById(R.id.btnChoseType);
84         btnDate = (Button) findViewById(R.id.btnDate);
85         btnSave = (Button) findViewById(R.id.btnSave);
86         btnQuery = (Button) findViewById(R.id.btnQuery);
87         btnDelete = (Button) findViewById(R.id.btnDelete);
88         btnCloseApp = (Button) findViewById(R.id.btnCloseApp);
89         btnClearMainOutput = (Button) findViewById(R.id.btnClearMainOutput);
90         btnChoseStartDate = (Button) findViewById(R.id.btnChoseStartDate);
91         btnChoseEndDate = (Button) findViewById(R.id.btnChoseEndDate);
92         txtMainOutput=(TextView) findViewById(R.id.txtMainOutput);
93         txtDate=(TextView) findViewById(R.id.txtDate);
94         txtStartDate=(TextView) findViewById(R.id.txtStartDate);
95         txtEndDate=(TextView) findViewById(R.id.txtEndDate);
96
97         // 建立一個SQL資料庫 (MyTallybook)
98         dbHelper = new DBHelper(this); //建立資料庫幫助物件
99         db = dbHelper.getWritableDatabase(); //透過幫助物件建立一個可以讀寫的資料庫
100
101         // 若是資料庫中沒有任何資料時，則在資料庫中建立預設的資料種類
102         Cursor cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE1, null);
103         if(cursor!=null)
104         {
105             int n = cursor.getCount();
106         }
107     }
108 }
```

```

105         if(n==0)
106         {
107             for (int i=0; i<initialExpenseTypes.length; i++)
108                 db.execSQL("INSERT INTO " + DATABASE_TABLE2 + " (expenseType) VALUES ('" +
109                     initialExpenseTypes[i] + "')");
110         }
111     }
112 }
113 //
114 @Override
115 protected void onStop()
116 {
117     super.onStop();
118     db.close(); // 關閉資料庫
119 }
120
121
122
123 // 點選選擇運動型態按鈕之事件處理程序
124 public void btnChooseType_Click(View view)
125 {
126     txtMainOutput.setText("");
127
128     Cursor cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2, null);
129     if(cursor != null)
130     {
131         int n = cursor.getCount();
132         expenseTypes = new String[n];
133         cursor.moveToFirst();
134         for (int i = 0; i < n; i++)
135         {
136             expenseTypes[i] = cursor.getString(1);
137             cursor.moveToNext();
138         }
139     }
140     else
141     {
142         expenseTypes = new String[1];
143         expenseTypes[0] = "";
144     }
145 }
146 // 建立運動型態單選對話方塊
147 AlertDialog.Builder builder = new AlertDialog.Builder(this);
148 builder.setTitle("選擇運動型態");
149 builder.setItems(expenseTypes, listenerChooseType);
150 AlertDialog dialogChooseType = builder.create();
151 dialogChooseType.show();
152 }
153 // 建立運動型態單選對話方塊之點擊事件處理程序
154 DialogInterface.OnClickListener listenerChooseType = new DialogInterface.OnClickListener()
155 {
156     public void onClick(DialogInterface dialog, int which)
157     {
158         btnChooseType.setText(expenseTypes[which]);
159         ExpenseType=expenseTypes[which];
160     }
161 };
162 // 管理運動型態對話之點擊事件處理程序
163 public void btnManageExpenseType_Click(View view)
164 {
165     txtMainOutput.setText("");
166
167     ManageExpenseTypeDialog manageExpenseTypeDialog = new ManageExpenseTypeDialog(this);
168     manageExpenseTypeDialog.setTitle("管理運動型態對話方塊");
169     manageExpenseTypeDialog.show();
170 }
171
172
173
174 // 選擇運動日期對話之點擊事件處理程序
175 public void btnDate_Click(View view)
176 {
177     txtMainOutput.setText("");
178     Calendar dt = Calendar.getInstance();
179     DatePickerDialog dDialog = new DatePickerDialog(this, new DatePickerDialog.OnDateSetListener() {
180         // 當點選日期對話的事件處理方法onDateSet()，所傳入的是所選擇的年、月、日
181         // 注意：月份的索引值從0開始，因此正確的月份為 (monthOfYear+1)
182         @Override
183         public void onDateSet(DatePicker view, int year, int monthOfYear, int dayOfMonth) //傳入使用者所選的年、月、日
184         {
185             PayDateString = year + "-" + (monthOfYear+1) + "-" + dayOfMonth;
186             txtDate.setText(year+"年"+ (monthOfYear+1) + "月" + dayOfMonth+ "日");
187             dt.set(Calendar.YEAR), dt.set(Calendar.MONTH),dt.set(Calendar.DAY_OF_MONTH);
188             dDialog.show();
189         }
190     });
191 }
192 // 儲存運動紀錄對話之點擊事件處理程序
193 public void btnSave_Click(View view)
194 {
195     txtMainOutput.setText("");
196     String priceString =txtPrice.getText().toString();
197     Comment=txtComment.getText().toString();
198
199     int empty=0;
200     if(priceString.length()==0)
201     {
202         Toast.makeText(this, "沒有填寫運動步數!", Toast.LENGTH_SHORT).show();
203         empty++;
204     }
205     if(ExpenseType.length()==0)
206     {
207         Toast.makeText(this, "選擇運動種類!", Toast.LENGTH_SHORT).show();
208         empty++;
209     }
210     if (PayDateString.length()==0)
211     {
212

```

```

210         Toast.makeText(this, "沒有選擇運動日期!", Toast.LENGTH_SHORT).show();
211         empty++;
212     }
213     if (empty==0)
214     {
215         try
216         {
217             Price=Integer.parseInt(priceString);
218
219             String commandString="INSERT INTO " + DATABASE_TABLE1 + " (price, expenseType, comment, payDate) VALUES " +
220                 "(" + Price + ", " + expenseType + ", " + Comment + ", " + PayDateString + ")";
221             db.execSQL(commandString);
222             Toast.makeText(this, "成功儲存新的運動紀錄!", Toast.LENGTH_SHORT).show();
223
224             etxtPrice.setText("");
225             etxtComment.setText("");
226             btnChoseType.setText("選擇運動種類");
227             txtDate.setText("");
228             PayDateString="";
229             ExpenseType="";
230         }
231         catch (Exception ex)
232         {
233             Toast.makeText(this, "程式出現例外: " + ex.getMessage(), Toast.LENGTH_SHORT).show();
234         }
235     }
236 }
237
238 // 查詢歷史運動紀錄列表之點擊事件處理程式
239 public void btnQuery_Click(View view)
240 {
241
242     txtMainOutput.setText("");
243
244     QueryHistoricalExpenseDialog queryHistoricalExpenseDialog = new QueryHistoricalExpenseDialog(this);
245     queryHistoricalExpenseDialog.setTitle("查詢運動紀錄對話方塊"); //設定標題
246     queryHistoricalExpenseDialog.show(); // 顯示對話方塊
247 }
248
249 // 清除顯示圖形列表之點擊事件處理程式
250 public void btnClearMainOutput_Click(View view)
251 {
252     txtMainOutput.setText(""); //清除主顯示區
253 }
254
255 // "關閉運動步數紀錄列表"之點擊事件處理程式
256 public void btnCloseApp_Click(View view)
257 {
258     #inish(); //關閉運動步數紀錄
259 }
260
261 }

```

圖 6

Branch: master PccuHW / 三下 / 智慧型手機程式設計 / Tallybook1_AlertDialogs / app / src / main / java / com / example / tallybook1_alertdialogs / ManageExpenseTypeDialog.java Find file Copy path

QINGSHURUNINDEXINGMING 2019/05/05, 17:24 e56fab0 · an hour ago 1 contributor

217 lines (192 sloc) | 8.7 KB Raw Blame History

```
1 //*****
2 /* 本類別實作管理消費種類之各個方法
3 /* by Min-Hsiung Hung 2014-04-19
4 //*****
5 package com.example.tallybook1_alertdialogs;
6
7 import android.app.AlertDialog;
8 import android.content.Context;
9 import android.database.Cursor;
10 import android.database.sqlite.SQLiteDatabase;
11 import android.view.View;
12 import android.widget.Button;
13 import android.widget.EditText;
14 import android.widget.TextView;
15 import android.widget.Toast;
16
17 public class ManageExpenseTypeDialog extends Dialog implements View.OnClickListener
18 {
19
20     // 與建立資料庫有關之變數宣告
21     private static String DATABASE_TABLE2 = "expenseTypes";
22     private SQLiteDatabase db;
23     private DBHelper dbHelper;
24
25     // 與顯示介面元件有關之變數宣告
26     Button btnCreateNewExpenseType, btnDeleteExpenseType;
27     Button btnDisplayExpenseTypes, btnCloseDialog;
28     EditText etxtNewExpenseType, etxtOldExpenseType;
29     TextView txtShowExpenseTypes;
30     Context myDialogContext;
31
32     public ManageExpenseTypeDialog(Context context) // 自訂對話方塊之建構子
33     {
34         super(context);
35
36         setContentView(R.layout.manageexpensetypedialog);
37
38         // 取得使用者介面之上所有元件
39         btnCreateNewExpenseType = (Button) findViewById(R.id.btnCreateNewExpenseType);
40         btnDeleteExpenseType = (Button) findViewById(R.id.btnDeleteExpenseType);
41         btnDisplayExpenseTypes = (Button) findViewById(R.id.btnDisplayExpenseTypes);
42         btnCloseDialog = (Button) findViewById(R.id.btnCloseDialog);
43         etxtNewExpenseType = (EditText) findViewById(R.id.etxtNewExpenseType);
44         etxtOldExpenseType = (EditText) findViewById(R.id.etxtOldExpenseType);
45         txtShowExpenseTypes = (TextView) findViewById(R.id.txtShowExpenseTypes);
46         myDialogContext = getContext(); // 取得對話方塊之管理物件
47
48         // 建立一個資料庫 (MyTallyBook)
49         dbHelper = new DBHelper(myDialogContext); // 建立資料庫輔助類別物件
50         db = dbHelper.getWritableDatabase(); // 透過輔助類別物件建立一個可以讀寫的資料庫
51
52
53         Cursor cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2, null);
54         if (cursor != null)
55         {
56             int n = cursor.getCount(); // 取得消費種類之筆數
57             String str = "目前已建立 " + n + " 個運動型態:\n";
58             cursor.moveToFirst(); // 將指標移到第一筆紀錄
59             for (int i = 0; i < n; i++) // 利用迴圈讀取每一筆紀錄
60             {
61                 str += (i+1) + ": " + cursor.getString(1) + "\n"; // 讀取第1個欄位值(消費種類 expenseType)，然後串接到顯示字串 (str) 中
62                 cursor.moveToNext(); // 將指標移到下一筆紀錄
63             }
64             txtShowExpenseTypes.setText(str);
65         }
66         else
67         {
68             txtShowExpenseTypes.setText("目前還沒有建立任何運動型態!\n");
69         }
70
71         btnCreateNewExpenseType.setOnClickListener(this);
72         btnDeleteExpenseType.setOnClickListener(this);
73         btnDisplayExpenseTypes.setOnClickListener(this);
74         btnCloseDialog.setOnClickListener(this);
75     }
76
77     // Click事件處理方法
78     public void onClick(View v)
79     {
80         if (v == btnCreateNewExpenseType)
81         {
82             String newExpenseTypeString = etxtNewExpenseType.getText().toString().trim();
83
84             int empty=0;
85             if(newExpenseTypeString.length()==0)
86             {
87                 Toast.makeText(myDialogContext, "沒有填寫新運動型態!", Toast.LENGTH_SHORT).show();
88                 empty++;
89             }
90             if (empty==0)
91             {
92                 try
93                 {
94                     db.execSQL("INSERT INTO " + DATABASE_TABLE2 + " (" + DATABASE_TABLE2 + ", " + DATABASE_TABLE2 + ") VALUES (" + newExpenseTypeString + ", " + newExpenseTypeString + ");");
95                 }
96                 catch (Exception e)
97                 {
98                     Toast.makeText(myDialogContext, "新增運動型態失敗!", Toast.LENGTH_SHORT).show();
99                 }
100             }
101         }
102         else if (v == btnDeleteExpenseType)
103         {
104             String oldExpenseTypeString = etxtOldExpenseType.getText().toString().trim();
105             if(oldExpenseTypeString.length()==0)
106             {
107                 Toast.makeText(myDialogContext, "沒有填寫舊運動型態!", Toast.LENGTH_SHORT).show();
108             }
109             else
110             {
111                 try
112                 {
113                     db.execSQL("DELETE FROM " + DATABASE_TABLE2 + " WHERE " + DATABASE_TABLE2 + " = '" + oldExpenseTypeString + "';");
114                 }
115                 catch (Exception e)
116                 {
117                     Toast.makeText(myDialogContext, "刪除運動型態失敗!", Toast.LENGTH_SHORT).show();
118                 }
119             }
120         }
121         else if (v == btnDisplayExpenseTypes)
122         {
123             Cursor cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2, null);
124             if (cursor != null)
125             {
126                 int n = cursor.getCount();
127                 String str = "目前已建立 " + n + " 個運動型態:\n";
128                 cursor.moveToFirst();
129                 for (int i = 0; i < n; i++)
130                 {
131                     str += (i+1) + ": " + cursor.getString(1) + "\n";
132                     cursor.moveToNext();
133                 }
134                 txtShowExpenseTypes.setText(str);
135             }
136             else
137             {
138                 txtShowExpenseTypes.setText("目前還沒有建立任何運動型態!\n");
139             }
140         }
141         else if (v == btnCloseDialog)
142         {
143             dismiss();
144         }
145     }
146 }
```

```

34
35
36 Cursor cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2 + " WHERE expenseType = '" +
37     newExpenseTypeString + "'", null);
38 int count = cursor.getCount();
39 if(count==0)
40 {
41     db.execSQL("INSERT INTO " + DATABASE_TABLE2 + " (expenseType) VALUES ('" +
42         newExpenseTypeString + "')");
43
44     Toast.makeText(myDialogContext, "新增運動型態成功!", Toast.LENGTH_SHORT).show();
45     etxtNewExpenseType.setText("");
46
47     cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2, null);
48     int n = cursor.getCount();
49     String str="目前已建立之運動型態:\n";
50     cursor.moveToFirst();
51     for (int i = 0; i < n; i++)
52     {
53         str += (i+1) + ": " + cursor.getString(i) + "\n";
54         cursor.moveToNext();
55     }
56     txtShowExpenseTypes.setText(str);
57 }
58 else
59 {
60     Toast.makeText(myDialogContext, "該運動型態已經存在!", Toast.LENGTH_SHORT).show();
61 }
62 }
63 catch(Exception ex)
64 {
65     Toast.makeText(myDialogContext, "程式出現例外: " + ex.getMessage(), Toast.LENGTH_SHORT).show();
66 }
67 }
68 //
69 if(v == btnDeleteExpenseType)
70 {
71     String oldExpenseTypeIDString = etxtOldExpenseType.getText().toString().trim();
72     int empty=0;
73     if(oldExpenseTypeIDString.length() == 0)
74     {
75         Toast.makeText(myDialogContext, "請再填寫運動型態編號!", Toast.LENGTH_SHORT).show();
76         empty++;
77     }
78     if (empty==0)
79     {
80         try
81         {
82             int oldExpenseTypeID = Integer.parseInt(oldExpenseTypeIDString);
83
84             Cursor cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2, null);
85             int count = cursor.getCount(); //取得紀錄筆數
86             if((oldExpenseTypeID > count) || (oldExpenseTypeID < 1))
87             {
88                 Toast.makeText(myDialogContext, "運動型態編號錯誤請重新輸入!", Toast.LENGTH_SHORT).show();
89             }
90             else
91             {
92                 cursor.moveToPosition(oldExpenseTypeID-1);
93                 String str2=cursor.getString(1);
94
95                 db.execSQL("DELETE FROM " + DATABASE_TABLE2 + " WHERE expenseType='" + str2 + "'");
96
97                 Toast.makeText(myDialogContext, "刪除運動型態成功!", Toast.LENGTH_SHORT).show();
98                 etxtOldExpenseType.setText("");
99
100                 cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2, null);
101                 if(cursor!=null)
102                 {
103                     int n = cursor.getCount();
104                     String str="目前已建立之運動型態:\n";
105                     cursor.moveToFirst();
106                     for (int i = 0; i < n; i++)
107                     {
108                         str += (i+1) + ": " + cursor.getString(i) + "\n";
109                         cursor.moveToNext();
110                     }
111                     txtShowExpenseTypes.setText(str);
112                 }
113             }
114         }
115         else
116         {
117             txtShowExpenseTypes.setText("目前並無任何運動型態!\n");
118         }
119     }
120 }

```

```

179         }
180     }
181     catch (Exception ex)
182     {
183         Toast.makeText(myDialogContext, "程式出現例外: " + ex.getMessage(), Toast.LENGTH_SHORT).show();
184     }
185 }
186 //
187 if (v == btnDisplayExpenseTypes)
188 {
189     Cursor cursor = db.rawQuery("SELECT * FROM " + DATABASE_TABLE2, null);
190     if (cursor != null)
191     {
192         int n = cursor.getCount(); //取得資料總數
193         String str = "目前已建立之運動型態:\n";
194         cursor.moveToFirst();
195         for (int i = 0; i < n; i++)
196         {
197             str += (i+1) + ": " + cursor.getString(1) + "\n";
198             cursor.moveToNext();
199         }
200         txtShowExpenseTypes.setText(str);
201     }
202     else
203     {
204         txtShowExpenseTypes.setText("目前尚未建立任何運動型態!\n");
205     }
206 }
207 }
208 }
209 //
210 if (v == btnCloseDialog) //點選關閉單件事件為"關閉對話方塊按鈕"，則進行以下處理
211 {
212     dismiss(); //關閉對話方塊
213 }
214 }
215 }
216 }

```

Branch: master PccuHW / 三下 / 智慧型手機程式設計 / Tallybook1_AlertDialogs / app / src / main / java / com / example / tallybook1_alertdialogs / QueryHistoricalExpenseDialog.java Find file Copy path

QINGSHURUNINDEXINGMING 2019/05/05_17:24 e56fab8 · an hour ago 1 contributor

286 lines (265 sloc) | 15.8 KB Raw Blame History

```
1  /**
2   * 本檔案製作查詢歷史消費紀錄之各個方法
3   *  By Hsin-Hsiung Hung 2014-04-19
4   */
5   package com.example.tallybook1_alertdialogs;
6
7   import java.util.Calendar;
8   import android.app.AlertDialog;
9   import android.app.DatePickerDialog;
10  import android.app.DatePickerDialog;
11  import android.app.DatePickerDialog;
12  import android.app.DatePickerDialog;
13  import android.app.DatePickerDialog;
14  import android.app.DatePickerDialog;
15  import android.app.DatePickerDialog;
16  import android.app.DatePickerDialog;
17  import android.app.DatePickerDialog;
18  import android.app.DatePickerDialog;
19  import android.app.DatePickerDialog;
20  import android.app.DatePickerDialog;
21
22  public class QueryHistoricalExpenseDialog extends Dialog implements View.OnClickListener
23  {
24
25      // 與建立資料庫有關之變數宣告
26      private static String DATABASE_TABLE1 = "tallybook";
27      private SQLiteDatabase db;
28      private DBHelper dbHelper;
29
30      // 與顯示介面元件相關之變數宣告
31      Button btnChoseStartDate, btnChoseEndDate;
32      Button btnQuery, btnDelete;
33      Button btnClearMainOutput, btnCloseDialog;
34      RadioButton radioButtonByDate, radioButtonByType;
35      TextView txtStartDate, txtEndDate, txtMainOutput;
36      Context myDialogContext;
37
38      // 清除螢幕
39      String startDateString="", endDateString="";
40
41      public QueryHistoricalExpenseDialog(Context context)
42      {
43          super(context);
44
45          setContentView(R.layout.queryhistoricalexpensesdialog);
46
47          // 取得畫面各個介面之上所有元件
48          btnChoseStartDate = (Button) findViewById(R.id.btnChoseStartDate);
49          btnChoseEndDate = (Button) findViewById(R.id.btnChoseEndDate);
50          btnQuery = (Button) findViewById(R.id.btnQuery);
51          btnDelete = (Button) findViewById(R.id.btnDelete);
52          btnClearMainOutput = (Button) findViewById(R.id.btnClearMainOutput);
53          btnCloseDialog = (Button) findViewById(R.id.btnCloseDialog);
54          radioButtonByDate = (RadioButton) findViewById(R.id.radioButtonByDate);
55          radioButtonByType = (RadioButton) findViewById(R.id.radioButtonByType);
56          txtStartDate = (TextView) findViewById(R.id.txtStartDate);
57          txtEndDate = (TextView) findViewById(R.id.txtEndDate);
58          txtMainOutput = (TextView) findViewById(R.id.txtMainOutput);
59          myDialogContext = getContext();
60
61          // 建立一個SQL資料庫 (MyTallyBook)
62          dbHelper = new DBHelper(myDialogContext);
63          db = dbHelper.getWritableDatabase(); // 透過輔助類別物件建立一個可以讀寫的資料庫
64
65          // 清除主顯示區域
66          txtMainOutput.setText("");
67
68          btnChoseStartDate.setOnClickListener(this);
69          btnChoseEndDate.setOnClickListener(this);
70          btnQuery.setOnClickListener(this);
71          btnDelete.setOnClickListener(this);
72          btnClearMainOutput.setOnClickListener(this);
73          btnCloseDialog.setOnClickListener(this);
74      }
75  }
```

```

76 }
77 // Click事件處理方法
78 public void onClick(View v) // 對該方法上8個操作按鈕之點擊事件都是由本方法來處理，傳入的物件為發出點擊事件之按鈕
79 {
80     if(v==btnChoseStartDate)
81     {
82         txtMainOutput.setText(""); //清除主顯示區
83         Calendar dt = Calendar.getInstance(); //取得一個日期物件
84         DatePickerDialog dDialog = new DatePickerDialog(myDialogContext, new DatePickerDialog.OnDateSetListener() {
85             // 當選擇日期後的事件處理方法onDateSet()，所傳入的是所選擇的年、月、日
86             // 注意：月份的索引是從0開始，因此正確的月份為(monthOfYear+1)
87             @Override
88             public void onDateSet(DatePicker view, int year, int monthOfYear, int dayOfMonth)
89             {
90                 StartDateStr = year + "-" + (monthOfYear+1) + "-" + dayOfMonth; //以所選日期建立一個格式化的起始日期字串
91                 txtStartDate.setText(year + "-" + (monthOfYear+1) + "-" + dayOfMonth + "日"); //將所選日期顯示在標題上
92                 dt.set(Calendar.YEAR), dt.set(Calendar.MONTH), dt.set(Calendar.DAY_OF_MONTH); //以目前日期為現日期
93             }
94         }, dt.get(Calendar.YEAR), dt.get(Calendar.MONTH), dt.get(Calendar.DAY_OF_MONTH)); //顯示日期選擇器對話方塊
95     }
96     //
97     if(v==btnChoseEndDate) //點選該點擊事件者為“選擇結束日期按鈕”，則執行以下處理
98     {
99         txtMainOutput.setText(""); //清除主顯示區
100         Calendar dt = Calendar.getInstance(); //取得一個日期物件
101         DatePickerDialog dDialog = new DatePickerDialog(myDialogContext, new DatePickerDialog.OnDateSetListener() {
102             // 當選擇日期後的事件處理方法onDateSet()，所傳入的是所選擇的年、月、日
103             // 注意：月份的索引是從0開始，因此正確的月份為(monthOfYear+1)
104             @Override
105             public void onDateSet(DatePicker view, int year, int monthOfYear, int dayOfMonth)
106             {
107                 EndDateStr = year + "-" + (monthOfYear+1) + "-" + dayOfMonth; //以所選日期建立一個格式化的結束日期字串
108                 txtEndDate.setText(year + "-" + (monthOfYear+1) + "-" + dayOfMonth + "日"); //將所選日期顯示在標題上
109                 dt.set(Calendar.YEAR), dt.set(Calendar.MONTH), dt.set(Calendar.DAY_OF_MONTH); //以目前日期為現日期
110             }
111         }, dt.get(Calendar.YEAR), dt.get(Calendar.MONTH), dt.get(Calendar.DAY_OF_MONTH)); //顯示日期選擇器對話方塊
112     }
113     //
114     if(v==btnQuery)
115     {
116         txtMainOutput.setText(""); //清除主顯示區
117         // 以下圖解使用含有選擇起始日期與結束日期
118         int empty=0;
119         if(StartDateStr.length()==0) // 若沒有選擇起始日期，則利用吐司訊息提醒使用者
120         {
121             Toast.makeText(myDialogContext, "請選擇起始日期!", Toast.LENGTH_SHORT).show();
122             empty++;
123         }
124         if(EndDateStr.length()==0) // 若沒有選擇結束日期，則利用吐司訊息提醒使用者
125         {
126             Toast.makeText(myDialogContext, "請選擇結束日期!", Toast.LENGTH_SHORT).show();
127             empty++;
128         }
129         if(empty==0) // 若使用者含有選擇起始日期及結束日期，則執行以下處理
130         {
131             try //利用try()執行可能產生例外的處理程式碼(如存取資料庫檔案)，該程式碼原有例外，也不會當掉，以增加app之強韌性
132             {
133                 if(radioButtonDate.isChecked()) // 點選日期日期先查詢
134                 {
135                     String commandStr = "SELECT * FROM " + DATABASE_TABLE1 + " WHERE payDate between '" +
136                         StartDateStr + "' and '" + EndDateStr + "' ORDER BY payDate";
137                     Cursor cursor = db.rawQuery(commandStr, null); //執行資料庫查詢命令
138                     if(cursor != null)
139                     {
140                         int n = cursor.getCount(); //取得資料筆數
141                         String str = "在" + StartDateStr + "到" + EndDateStr + "共有" + n + "筆運動紀錄:\n";
142                         int totalAmount = 0;
143                         cursor.moveToFirst(); //將指標移到第1筆紀錄
144                         for (int i = 0; i < n; i++) // 利用迴圈逐一讀取每一筆紀錄
145                         {
146                             totalAmount += Integer.parseInt(cursor.getString(1));
147                             cursor.moveToNext(); // 移動到下一筆
148                         }
149                         str += "共計消費 " + totalAmount + " 元!\n";
150                         // 顯示消費紀錄之每一個單位之名稱
151                         String[] colNames = {"編號", "步數", "運動型態", "說明", "運動日期"};
152                         for (int i = 0; i < colNames.length; i++)
153                             str += String.format("%s\t", colNames[i]); // 將每一個單位的名稱串接到顯示字串(str)中
154                         str += "\n";
155                     }
156                     cursor.moveToFirst(); // 將指標移到第1筆紀錄
157                     // 顯示單位
158                     for (int i = 0; i < n; i++) //利用迴圈讀取每一筆紀錄之各個單位
159                     {
160                         str += String.format("%s\t", (i+1)); // 串接記錄編號(索引值-1)
161                         str += String.format("%s\t", cursor.getString(1));
162                         str += String.format("%s\t", cursor.getString(2));
163                         str += String.format("%s\t", cursor.getString(3));
164                         str += String.format("%s\t", cursor.getString(4));
165                         str += "\n";
166                         cursor.moveToNext(); // 移動到下一筆
167                     }
168                     txtMainOutput.setText(str);
169                 }
170                 else
171                 {
172                     txtMainOutput.setText("在" + StartDateStr + "到" + EndDateStr + "並無運動紀錄!\n");
173                 }
174             }
175             if(radioButtonType.isChecked())
176             {
177                 Cursor cursor = db.rawQuery("SELECT DISTINCT expenseType FROM " + DATABASE_TABLE1 +
178                     " WHERE payDate between '" + StartDateStr + "' and '" +
179                     EndDateStr + "'", null); //執行資料庫查詢命令
180                 if(cursor != null)
181                 {
182                     String str1 = "在" + StartDateStr + "到" + EndDateStr + "之步數統計如下:\n";
183                     String str="";
184                     int n = cursor.getCount();
185                     int totalAmount=0, num1; //用於紀錄
186                     String [] types = new String[n];
187                     cursor.moveToFirst(); // 移動到第1筆紀錄
188                     for(int i=0; i<n; i++)
189                     {
190                         types[i]=cursor.getString(0);
191                     }
192                 }
193             }
194         }
195     }

```

```

190
191
192         String commandString = "SELECT SUM(price) FROM " + DATABASE_TABLE1 +
193             " WHERE (payDate between '" + StartDateString + "' AND '" +
194             EndDateString + "') AND (expenseType = '" + types[i] + "')";
195         Cursor cursor1 = db.rawQuery(commandString, null);
196         cursor1.moveToFirst();
197         if(cursor1 != null)
198             num1 = cursor1.getInt(0);
199         else
200             num1=0;
201         totalAmount += num1;
202         str += String.format("$$$: " + "$$s \n", types[i], num1);
203         cursor.moveToNext();
204     }
205     str1 = "運動步數總計" + totalAmount + "\n";
206     str1 += str;
207     txtMainOutput.setText(str1); //將顯示字串str1的內容顯示在主顯示區文字區上
208 }
209 else // 若有連續消費紀錄，則顯示沒有消費紀錄
210 {
211     txtMainOutput.setText("在" + StartDateString + "到" + EndDateString + "並沒有運動紀錄!\n");
212 }
213 }
214 }
215 catch(Exception ex) // try()中的程式碼執行時發生例外，則利用catch去捕捉，並利用吐司訊息提示給使用者知道
216 {
217     Toast.makeText(myDialogContext, "程式出現例外: " + ex.getMessage(), Toast.LENGTH_SHORT).show();
218 }
219 }
220 }
221 //
222 //
223 //
224 //
225 //
226 //
227 //
228 //
229 //
230 //
231 //
232 //
233 //
234 //
235 //
236 //
237 //
238 //
239 //
240 //
241 //
242 //
243 //
244 //
245 //
246 //
247 //
248 //
249 //
250 //
251 //
252 //
253 //
254 //
255 //
256 //
257 //
258 //
259 //
260 //
261 //
262 //
263 //
264 //
265 //
266 //
267 //
268 //
269 //
270 //
271 //
272 //
273 //
274 //
275 //
276 //
277 //
278 //
279 //
280 //
281 //
282 //
283 //
284 //
285 //
286 //
287 //
288 //
289 //
290 //
291 //
292 //
293 //
294 //
295 //
296 //
297 //
298 //
299 //
300 //
301 //
302 //
303 //
304 //
305 //
306 //
307 //
308 //
309 //
310 //
311 //
312 //
313 //
314 //
315 //
316 //
317 //
318 //
319 //
320 //
321 //
322 //
323 //
324 //
325 //
326 //
327 //
328 //
329 //
330 //
331 //
332 //
333 //
334 //
335 //
336 //
337 //
338 //
339 //
340 //
341 //
342 //
343 //
344 //
345 //
346 //
347 //
348 //
349 //
350 //
351 //
352 //
353 //
354 //
355 //
356 //
357 //
358 //
359 //
360 //
361 //
362 //
363 //
364 //
365 //
366 //
367 //
368 //
369 //
370 //
371 //
372 //
373 //
374 //
375 //
376 //
377 //
378 //
379 //
380 //
381 //
382 //
383 //
384 //
385 //
386 //
387 //
388 //
389 //
390 //
391 //
392 //
393 //
394 //
395 //
396 //
397 //
398 //
399 //
400 //
401 //
402 //
403 //
404 //
405 //
406 //
407 //
408 //
409 //
410 //
411 //
412 //
413 //
414 //
415 //
416 //
417 //
418 //
419 //
420 //
421 //
422 //
423 //
424 //
425 //
426 //
427 //
428 //
429 //
430 //
431 //
432 //
433 //
434 //
435 //
436 //
437 //
438 //
439 //
440 //
441 //
442 //
443 //
444 //
445 //
446 //
447 //
448 //
449 //
450 //
451 //
452 //
453 //
454 //
455 //
456 //
457 //
458 //
459 //
460 //
461 //
462 //
463 //
464 //
465 //
466 //
467 //
468 //
469 //
470 //
471 //
472 //
473 //
474 //
475 //
476 //
477 //
478 //
479 //
480 //
481 //
482 //
483 //
484 //
485 //
486 //
487 //
488 //
489 //
490 //
491 //
492 //
493 //
494 //
495 //
496 //
497 //
498 //
499 //
500 //
501 //
502 //
503 //
504 //
505 //
506 //
507 //
508 //
509 //
510 //
511 //
512 //
513 //
514 //
515 //
516 //
517 //
518 //
519 //
520 //
521 //
522 //
523 //
524 //
525 //
526 //
527 //
528 //
529 //
530 //
531 //
532 //
533 //
534 //
535 //
536 //
537 //
538 //
539 //
540 //
541 //
542 //
543 //
544 //
545 //
546 //
547 //
548 //
549 //
550 //
551 //
552 //
553 //
554 //
555 //
556 //
557 //
558 //
559 //
560 //
561 //
562 //
563 //
564 //
565 //
566 //
567 //
568 //
569 //
570 //
571 //
572 //
573 //
574 //
575 //
576 //
577 //
578 //
579 //
580 //
581 //
582 //
583 //
584 //
585 //
586 //
587 //
588 //
589 //
590 //
591 //
592 //
593 //
594 //
595 //
596 //
597 //
598 //
599 //
600 //
601 //
602 //
603 //
604 //
605 //
606 //
607 //
608 //
609 //
610 //
611 //
612 //
613 //
614 //
615 //
616 //
617 //
618 //
619 //
620 //
621 //
622 //
623 //
624 //
625 //
626 //
627 //
628 //
629 //
630 //
631 //
632 //
633 //
634 //
635 //
636 //
637 //
638 //
639 //
640 //
641 //
642 //
643 //
644 //
645 //
646 //
647 //
648 //
649 //
650 //
651 //
652 //
653 //
654 //
655 //
656 //
657 //
658 //
659 //
660 //
661 //
662 //
663 //
664 //
665 //
666 //
667 //
668 //
669 //
670 //
671 //
672 //
673 //
674 //
675 //
676 //
677 //
678 //
679 //
680 //
681 //
682 //
683 //
684 //
685 //
686 //
687 //
688 //
689 //
690 //
691 //
692 //
693 //
694 //
695 //
696 //
697 //
698 //
699 //
700 //
701 //
702 //
703 //
704 //
705 //
706 //
707 //
708 //
709 //
710 //
711 //
712 //
713 //
714 //
715 //
716 //
717 //
718 //
719 //
720 //
721 //
722 //
723 //
724 //
725 //
726 //
727 //
728 //
729 //
730 //
731 //
732 //
733 //
734 //
735 //
736 //
737 //
738 //
739 //
740 //
741 //
742 //
743 //
744 //
745 //
746 //
747 //
748 //
749 //
750 //
751 //
752 //
753 //
754 //
755 //
756 //
757 //
758 //
759 //
760 //
761 //
762 //
763 //
764 //
765 //
766 //
767 //
768 //
769 //
770 //
771 //
772 //
773 //
774 //
775 //
776 //
777 //
778 //
779 //
780 //
781 //
782 //
783 //
784 //
785 //
786 //
787 //
788 //
789 //
790 //
791 //
792 //
793 //
794 //
795 //
796 //
797 //
798 //
799 //
800 //
801 //
802 //
803 //
804 //
805 //
806 //
807 //
808 //
809 //
810 //
811 //
812 //
813 //
814 //
815 //
816 //
817 //
818 //
819 //
820 //
821 //
822 //
823 //
824 //
825 //
826 //
827 //
828 //
829 //
830 //
831 //
832 //
833 //
834 //
835 //
836 //
837 //
838 //
839 //
840 //
841 //
842 //
843 //
844 //
845 //
846 //
847 //
848 //
849 //
850 //
851 //
852 //
853 //
854 //
855 //
856 //
857 //
858 //
859 //
860 //
861 //
862 //
863 //
864 //
865 //
866 //
867 //
868 //
869 //
870 //
871 //
872 //
873 //
874 //
875 //
876 //
877 //
878 //
879 //
880 //
881 //
882 //
883 //
884 //
885 //
886 //
887 //
888 //
889 //
890 //
891 //
892 //
893 //
894 //
895 //
896 //
897 //
898 //
899 //
900 //
901 //
902 //
903 //
904 //
905 //
906 //
907 //
908 //
909 //
910 //
911 //
912 //
913 //
914 //
915 //
916 //
917 //
918 //
919 //
920 //
921 //
922 //
923 //
924 //
925 //
926 //
927 //
928 //
929 //
930 //
931 //
932 //
933 //
934 //
935 //
936 //
937 //
938 //
939 //
940 //
941 //
942 //
943 //
944 //
945 //
946 //
947 //
948 //
949 //
950 //
951 //
952 //
953 //
954 //
955 //
956 //
957 //
958 //
959 //
960 //
961 //
962 //
963 //
964 //
965 //
966 //
967 //
968 //
969 //
970 //
971 //
972 //
973 //
974 //
975 //
976 //
977 //
978 //
979 //
980 //
981 //
982 //
983 //
984 //
985 //
986 //
987 //
988 //
989 //
990 //
991 //
992 //
993 //
994 //
995 //
996 //
997 //
998 //
999 //
1000 //

```

圖 8

三、學習心得

心得感想

手機作業記錯繳交時間了，覺得好難過